Development of Tools to Represent Design Procedures

David A. Gebala
Steven D. Eppinger
Daniel E. Whitney

WP# 3281-91-MS

April 1991

Development of Tools to Represent Design Procedures

David A. Gebala
Steven D. Eppinger
Daniel E. Whitney

WP# 3281-91-MS April 1991

Development of Tools to Represent Design Procedures

David A. Gebala
Steven D. Eppinger
Daniel E. Whitney

Massachussetts Institute of Technology
Cambridge, MA 02139

Abstract

This paper is based upon the premise that in practice, design procedures are rather methodical and can therefore be studied and improved. We present model-based approaches toward understanding such design activities in order to improve design practices. Several modeling methodologies are presented and compared on their ability to capture relevant information about design activities and their ability to provide insight into the performance of design processes. A detailed description of these models and examples of each are provided for comparison. These models contribute significantly to our understanding of current design practice and yield to analytical tools which facilitate formulation and implementation of improved procedures. In the course of the discussion, limitations of the currently used models are identified, and an enhanced representation is proposed to overcome these limitations.

Introduction

International competition has heightened the need to develop high quality products which can compete in the global marketplace. As a result of this increased competition, the pace of product development has quickened, forcing manufacturers into an era in which continuous quality improvement is a matter of survival, not simply competitive advantage. As the time scale of the product life cycle decreases and the demand for quality increases, one area of attention has been improving products through better designs. The integrative and definitive nature of the design activity is responsible for determining as much as 80% of a product's functionality and cost [4]. The high leverage available during the design activity can completely determine a product's success or failure. Design's central role as an important engineering and business concern has recently gained attention in companies and universities around the world.

In the past, many researchers may have dismissed design because it lacks the scientific rigor of some other disciplines. However, as academia and industry invest more effort into improving the design process, there is an increased need for a better understanding of the activity to facilitate the formulation of new design paradigms. We believe that while most design

activities performed in industry are well established,they are, nevertheless, poorly understood.

We seek improved understanding of both new and established design procedures to provide at least two distinct opportunities: (1) to allow engineers to execute the design activity with more insight into difficulties encountered during design, and (2) to provide an opportunity for better management of the design activity, including implementation of new design strategies. Specifically, this work addresses design process improvement questions which engineers and their managers should be asking:

1) *Why does it take so long to develop each new generation of our product?*
2) *Which engineering functions might be combined to accelerate design progress?*
3) *Where is communication most important in design?*
4) *What are the driving factors in our design problem?*
5) *How can we implement concurrent engineering?*

The approach in this work is to facilitate the improvement of product design procedures through formulating and testing models of the design activity. Models are constructed by first observing the design process and formulating representations which capture the aspects considered important. A model serves its purpose if it is able to answer important questions about the subject it represents. If the model's behavior reflects the system's behavior, the model can be used in some predictive capacity to explore the effects which altering the system can produce. If the representation is not able to answer such questions, another model must be formulated and tested. The models surveyed below will be judged on their ability to accurately represent design activities and their ability to facilitate exploration of new design strategies.

In choosing a model of the design process, it is important to keep some objectives in mind. Winston proposes several requirements which problem representations should attempt to satisfy [24]. Good representations

- make the important things explicit;
- expose natural constraints, facilitating computations;
- are complete;
- are concise;
- are transparent to its users; and
- suppress detail when it is not required.

These criteria can serve as guidelines for representations which can advance our understanding of the complex activities which are collectively known as design.

The first section of this paper presents three models of the design activity commonly used in practice. We review the historical development of the Program Evaluation and Review Technique (PERT), Structured Analysis and Design Technique (SADT or IDEF), and the use of Design Structure Matrices (DSM). We also present examples of their use in design modeling and survey them critically to identify any shortcomings. In particular, we examine their ability to reveal the difficult iterations encountered in execution of the design activity.

In the following section, we compare the representational capacity of the two techniques best able to capture the iterative flow of design information. Using precedence data describing the design of an automotive component, alternate representations of a design process were constructed. We then use the precedence matrices to model several other design activities, and to provide valuable insight into how information flows during these design procedures. We close the section with an overview of analysis tools which are able to suggest improved design procedures.

Finally, we will address the issues which the representations are not able to model. Our work in modeling and analyzing design has identified important aspects of the design process which are not incorporated into any of the present models. For example, the existing models only represent interdependence, but do not specify the extent of interdependence. A possible solution is proposed to incorporate much more information into the models to enhance their utility. One type of quantitative model based on the precedence matrix is presented in an example.

Existing Representations

As the importance of product design in manufacturing competition becomes widely recognized, the need to understand and improve the design activities also becomes evident. The first attempts to model and answer questions about design procedures used simple project planning techniques (PERT) which had proven useful in the scheduling and management of large research and development projects. However, these simple tool did not contribute to understanding design iteration because they failed to represent such flows. This need to represent iteration explicitly was met by techniques such as SADT. Finally, we have found that precedence matrices such as DSM allow complex interactions to be shown more easily. These three most commonly used representations are discussed below.

1. Program Evaluation and Review Technique

The Program Evaluation and Review Technique (PERT) is one of the most widely used project management tools. PERT was developed in the late 1950's to plan and manage the development of the Polaris missile [23]. A project is represented as a series of activities, each with a set of predecessors whose results are required before the new activity can be started. The PERT chart which documents the project is a network of nodes interconnected by arcs. Although there is some flexibility in defining the nodes and arcs, it is most common to associate one activity with each node and to denote precedence using a directed arc between nodes. To add the time to the PERT chart, we ca label the arcs with the task durations. Uncertainties in these time estimates can be considered in the calculation of the overall project completion time. An example of a PERT chart with deterministic time estimates is presented in Figure 1.

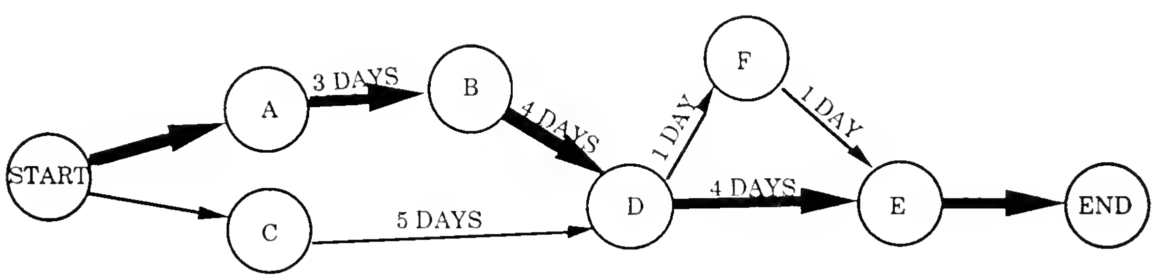

Figure 1. PERT Chart

If the time required for each of the activities has a small variability, the PERT chart can be subjected directly to Critical Path Method (CPM) analysis. CPM allows the manager to identify which activities affect the completion date (activities on the critical path) and which activities are irrelevant to accelerating project completion (activities with slack). By

analyzing the critical path, the manager can perform cost vs. time trade-off analysis [12]. The cost of applying more resources to an activity can be evaluated by the effect on the project completion date. The critical path is the one marked by heavy arcs in Figure 1.

The PERT methodology is well suited for certain processes which involve sequential flows of well characterized steps. For example, PERT is a popular planning tool for construction and assembly operations which typically involve little flexibility in their precedence relations. In order to make a particular subassembly, one needs all of its components. The strong dependency which is found in the assembly and construction activities is easily represented by the series of arcs and nodes in a PERT chart.

PERT charts have also found widespread application in large scale research and development projects requiring coordination of numerous activities. Again, the projects flow sequentially: before a certain task is activated, the previous task must be completed. The success of this technique in the development of the Polaris missile prompted the U.S. Government to require its use on all major government projects [23]. Using PERT to model the design activity seems to be a logical choice since design, too, is a large project requiring coordination of many efforts and many different activities.

Using PERT to Model Design Activities

Using PERT charts to model the design process seems a straightforward application of this technique. The design activity must be decomposed into smaller activities and their precedences noted. The duration of each activity can be used to enhance the chart's information content. However, the technique is not able to capture the complexities of the design activity; PERT is limited by the implicit requirement that all activities are sequentially dependent. Once an activity has been performed, it is not performed again. In a PERT chart, such an occurrence would create a cycle in the path. Such cycles in the PERT chart represent logical inconsistencies which must be removed before any critical path calculations can be made.

In practice design is often an iterative activity. Design tasks may not have strict start and stop dates. Rather, a design decision made downstream of an activity may force this upstream activity to be re-activated. The iterative nature of the design process also means that incomplete information may be used to perform an activity which will be performed again when more complete information becomes available. These cycles are common in most design activities and should not be ignored in our models. In fact, it is this iteration using incomplete

information which presents the challenge in design. If we choose to "force-fit" the design process into the PERT representation, we are forced to ignore these cycles, and instead of making the important things explicit, we have simplified design into a trivial sequence of straightforward activities. Clearly, any useful representation must be able to represent cycles in the design process explicitly.

2. Structured Analysis and Design Technique

Another popular technique which is used extensively in practice is the Structured Analysis and Design Technique (SADT). This method is a graphical means of describing large systems of interacting units. SADT shares some common background with PERT. Both were motivated by the need to coordinate a large number of activities which may occur at different times and at different locations. The derivative of SADT which is widely used today is one promoted by the U.S. Air Force in the early 1970's in an attempt to standardize manufacturing process descriptions across many different aerospace contractors [13]. The Integrated Computer-Aided Manufacturing Definition Method (IDEF0) is an automated, graphical adaptation of SADT aimed at standardizing contractor communications and reviews [17].

The SADT/IDEF0 technique is based on a basic unit, the Structured Analysis (SA) box, and a fundamental descriptive language which dictates how these boxes can be interrelated [16]. Figure 2 presents the SA building block of the IDEF0 models. Each box in the model represents an activity. An activity is defined as an act which, under control, transforms input into output using a mechanism. Each box can have any number of arrows leading into and out of it, following the convention pictured. Arrows directed into the box from the right represent input. Arrows directed out of the box represent output. The arrows entering the top of the activity box represent control. Those entering from the bottom represent mechanisms used, but not changed during the activity. Any SA box may be a component of another, higher-level SA box, and may itself be decomposable into more component boxes. The approach is to model from the top down, specifying as much detail, and creating as much depth as necessary to completely define the system being modeled.

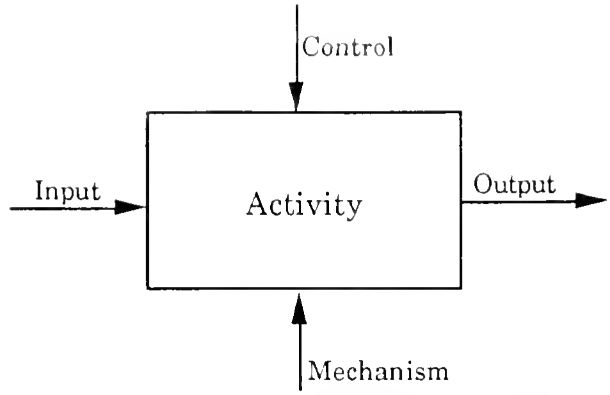

Figure 2. Structured Analysis (SA) Box

The interview is the main means of data collection, and is supplemented by reading documents and observing activities . Once the data has been collected, it is formatted and presented according to the IDEF0 modeling guidelines. To construct an IDEF0 document, the author applies the scripting language which guides the layout of the document [13]. This adherence to convention is the common language that allows communication between different parties. The reader of the document should be able to understand the interaction between activities based solely on the graphical content and SA labels (although text documents can be added for clarification). An example of an IDEF0 document is presented in Figure 3.

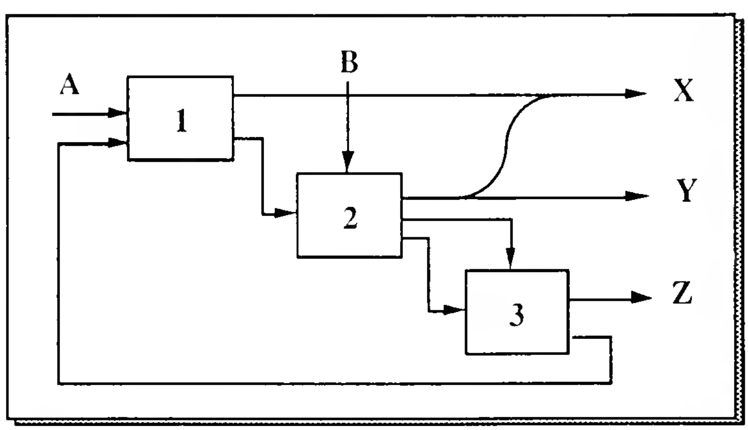

Figure 3. SADT/IDEF0 Document

Experts in the IDEF0 methodology stress that the representation is a living document which is iteratively updated and proofed until all parties have reached a consensus. Before a model is complete, it must be checked for omissions and validity. The people close to the activities being modeled are asked to verify the relationships represented in the document.

Necessary modifications are made and the cycle proceeds through a number of iterations before a model is approved.

Using SADT to Model Design Activities

Models of numerous design procedures have been constructed using the IDEF0 technique. To do so, the design activity to be modeled is hierarchically decomposed and documented on as many different levels as required. There is a strict limit to the amount of information that can be placed on each page of the IDEF0 document. This restriction is an attempt to limit the degree of complexity which the reader of the document is exposed to at any one time. The documents themselves consist of multiple sheets, with each page documenting all the inputs, outputs, controls and mechanisms which affect a particular set of activities. Note that these interrelations are often not from boxes at the same level of decomposition and are not necessarily on the same sheet of the IDEF0 document.

Most large manufacturing firms we know of, including Boeing, General Motors, Kodak and Motorola, have used the IDEF0 software and methodology to create models of existing design procedures. Often the tasks to be modeled are enormous and must be decomposed into functional areas. Each area is made responsible for collecting information and creating a valid IDEF0 document which can be used to study its existing design practice.

At General Motors,the complete process model for automobile design is still being constructed, but some parts have been completed. The resulting documents represent huge investments in resources, and are rich in information; however, in practice, the size of these documents has proven to be one of the most significant drawbacks. Because the software does not perform it automatically, verification of interrelations between activities requires much effort tracing arrows backwards and forwards through multiple levels of decomposition. Because the overall design activity is large and complex, the documents themselves become a tangled network which is laborious to read.

The IDEF0 procedure is indeed able to capture the complex requirements of the design process, but the result does not lend insight into the design process. This representation of the design activity serves as a documentation of the design process, but because of its complexity and unwieldy size, it is not particularly good at suggesting improvements. The IDEF0 document does not facilitate computation nor does it make difficulties explicit. This shortcoming renders it inadequate for improving the design process. A useful model must be able to answer questions with much less effort than IDEF0 models require.

3. Matrix Representations

Beginning in the 1960's, some researchers have used matrix representations in place of graphs to display systems of equations and precedences. Adjacency matrices are square matrices whose elements, a_{ij}, are unity if node j is a precedent of node i, and blank otherwise [22]. The matrix is easily constructed from the network diagrams or directly from precedence information. The size of the matrix depends upon the number of activities or tasks into which the system is broken. The number of non-blank elements reflects the number of precedences in the system. Figure 4 depicts a typical adjacency matrix. One would read the precedence relations from the matrix by reading across the rows of the matrix. For example, in Figure 4, Task D is dependent upon input from Tasks A and E. The name, adjacency matrix, reflects the fact that the non-blank elements identify which nodes are adjacent (connected) to each other [8]. No special language is required to construct the matrix, and the results are easily understood by both the creator and the user.

	A	B	C	D	E	F	G	H	I	J	K	L
A									1			
B							1		1	1		1
C						1			1			
D	1				1							
E			1					1				
F			1									
G		1			1							1
H				1								
I										1		
J	1											
K						1						
L	1						1				1	

Figure 4. Adjacency Matrix

The matrix representation has been used to model many different processes and systems. Mathematicians have used this matrix form to solve simultaneous equations [11, 22]. Chemical engineers have used the matrix representation to model process flow sheets [9]. Engineers have documented the interdependence to examine the most efficient order in which to determine system parameters [2, 3, 15, 20].

Using Precedence Matrices to Model Design Activities

In a similar manner, matrix representations have been used to model the information flow in the design process. The same precedence

information used in the other techniques is collected for each component task. If one interprets the task ordering as a time sequence, the timing of information flow becomes explicit: marks below the diagonal represent information transferred to later tasks; marks above the diagonal depict information fed back to earlier tasks. The matrix embodies the structure of the underlying design activity by mapping the relations between tasks in a precise order which makes interdependence explicit. Steward refers to a matrix model of design activities as a Design Structure Matrix (DSM) [20].

This representation has been used in a number of design projects to successfully map dependencies between functional areas of the design endeavor. The work done by Black exposes, on a parameter level, the information requirements of brake system design [2]. This work allows the designers to see the entire design activity from a global perspective. The order in which parameters are determined can have a significant effect on the execution of the design and these features are revealed in the matrix. The details of this work are presented in the next section.

Most recently we have used the matrix representation as a management aid to analyze the organizational structure of design projects using the individual information requirements of the engineering tasks [6, 19]. As an aid to better design management, we have used the matrix to identify tasks which are constrained to be serial by nature of the information flow connecting them. Conversely, the matrix also allows quick identification of tasks which can be performed in parallel. This representation has proven useful in studying the changes required in the implementation of concurrent engineering strategies [5]. These results are discussed further in the final section of this paper.

Comparison of Representations

To compare the design models constructed by the different techniques, we modeled design activities using the IDEF0 and matrix methods presented above. We found that the IDEF0 methodology has been applied extensively to represent the design and manufacturing activities at General Motors. The C4 program at GM has formed Lead Groups which are responsible for documenting the design procedures for each of the seven major functional areas of the automobile: 1) Powertrain, 2) Chassis, 3) Electrical, 4) Body-in-White, 5) Paint, 6) Assembly and 7) Interior. The aim of this endeavor is to understand the design and production activities as they are currently performed and to then suggest improvements for future performance.

The precedence information gathered while studying and documenting the production of these components is valuable information which could, if represented and analyzed correctly, aid in improving and accelerating the design activity. At present, all of this precedence information for the activities is contained in IDEF0 charts. These charts are constructed by authors who interview engineers in the organizations, are reviewed for completeness and validity, and are eventually approved as an accurate representation of the current design practice at General Motors.

These IDEF0 models are large and complex, reflecting the processes they model. However, the large size of the models is a disadvantage of the representation. The size and complexity of the documents are barriers to their use in facilitating improvements of the processes they represent. Although the description is complete, it is not concise enough to serve as a beneficial overview tool. This representation fails to suppresses enough detail to render it transparent to users. The vast amount of information presented only one page at a time provides only limited glimpses into the global interactions of design and manufacturing which are often crucial.

The same precedence data used in the IDEF0 charts can be mapped into the matrix format and compared to the SA box notation. Figure 5 is a schematic of the translation of IDEF0 data into its equivalent matrix

format.

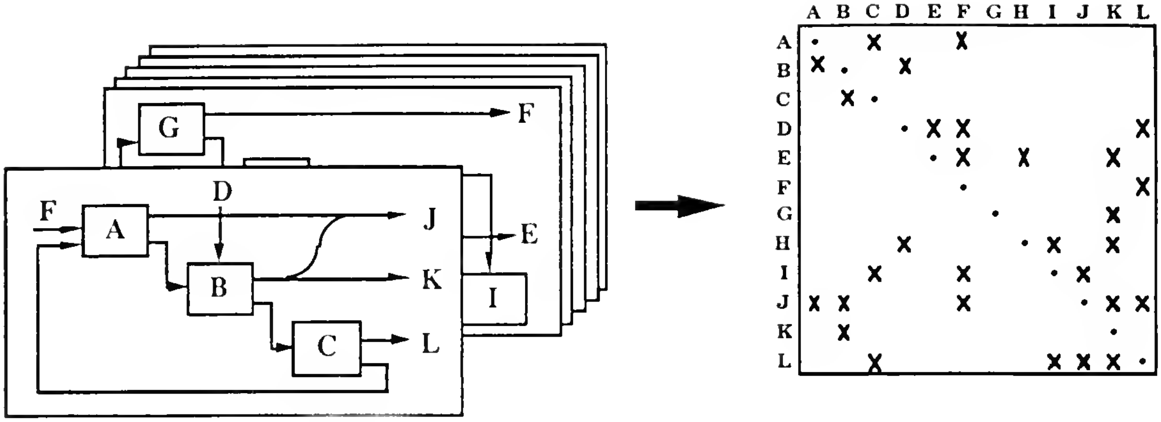

Figure 5. Translation from IDEF0 Format to Matrix Format

Each lowest level SA activity box becomes a row in the matrix. We have performed this transformation for only two design problems in the power train lead group (a crankshaft and a camshaft), but the results have provided interesting comparisons to the IDEF0 format. The matrix is also large, but it is an orderly array in which precedent information is obvious for each of the tasks included in the model. We found the matrix representation much easier to construct and read. Because the matrix representation is easier to verify, it is better suited to accurately represent current practice as well as provide suggestions for improvements.

Three different models were constructed based on data from General Motors. The first activity presented below is the brake system design. In this study parameter level data was collected to document the interaction between physical design parameters such as energetic,geometric, and material properties. This is a model of the engineering interactions which must be considered in the design of a brake system. In contrast, the second model presents data at a task level. The precedence data for this model reveals how the separate tasks of the crankshaft design activity must interact with one another during the design process. A more comprehensive view is offered in the third model which is based upon the entire camshaft development process including activities from design through production. Each of these models has contributed to our understanding of the methodology of design and will be discussed individually in the following section.

Brake System Design

The first study was performed by Tom Black on brake system design [2]. The model was constructed from data collected during interviews and discussions with the designers. Designers were asked to identify what information was required to perform the specific jobs. The jobs were

detailed down to the parameter level to include measurable quantities such as pressure, diameter, and torque. The model was constructed and verified in the matrix form pictured in Figure 6A.

Notice that the design task is decomposable into three distinct phases. The information flow in the early design stage is a mixture of sequential and parallel activities. In the second phase of the design activity, the information flow becomes iterative as represented by the large block of tasks which are coupled. Following this block is final group of sequential and parallel tasks. The decisions which are resolved iteratively in the middle of the design process are presented in Figure 6B. These tasks could be analyzed in further detail using the methods described in [7, 20]. Some of these were performed and the results are reported in [2, 21].

Certain aspects of this model should be noted. The information flow during certain stages of design is critical while during much of the activity, it is a simple sequence. The coupled block of iterative information flow in the middle of the design matrix is what we consider the challenge of the design activity. We expect that all but the most trivial design activities will have some degree of iteration. We investigate below whether this overall structure is common to other design activities.

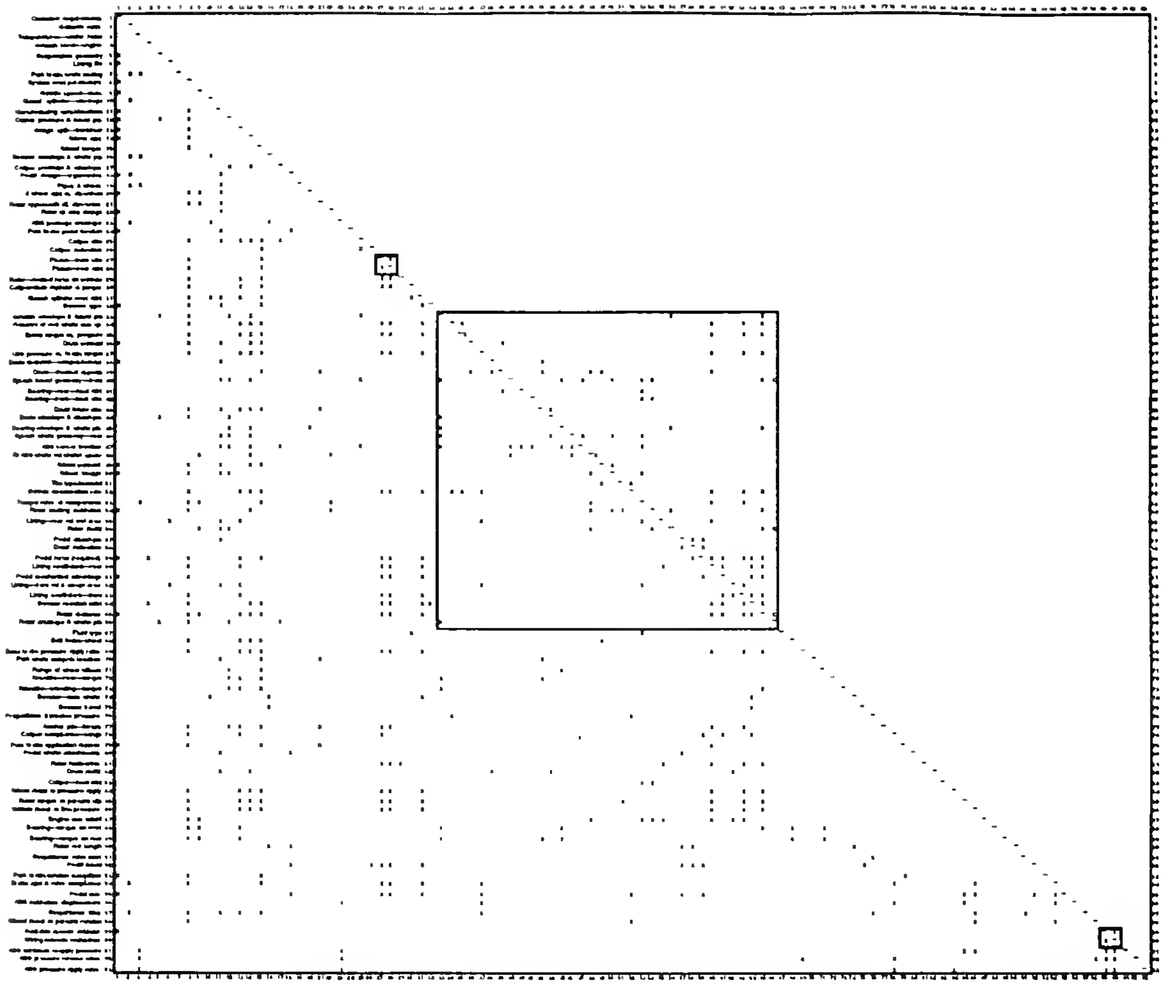

Figure 6A. Brake System Matrix Model

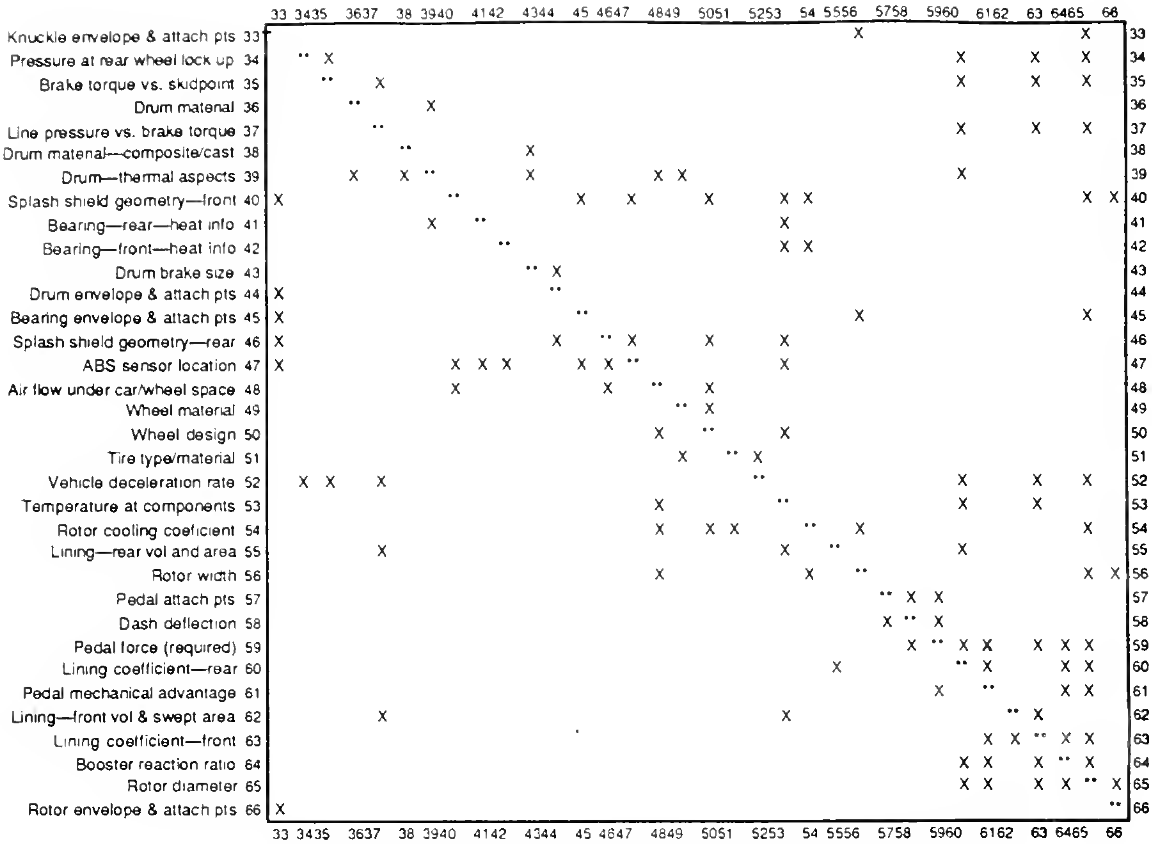

Figure 6B. Iterative Block of Brake System Matrix Model

Crankshaft Design

A second design process model was produced from the study of crankshaft design. This model was developed by GM's C4 Power Train Lead Group using the IDEF0 methodology. The information they gathered was subsequently translated into the matrix format. The original IDEF0 document consists of more than 20 pages and is too large to reproduce here. However, the matrix representation is provided in Figure 7. The relation assigned during the IDEF0 modelling has been preserved: Input/Output, Control, and Mechanism are denoted in the matrix as I, C, and M, respectively. Translating the precedence relations from the IDEF0 document into the matrix provided a good overview of the design activity as captured by the IDEF0 authors.

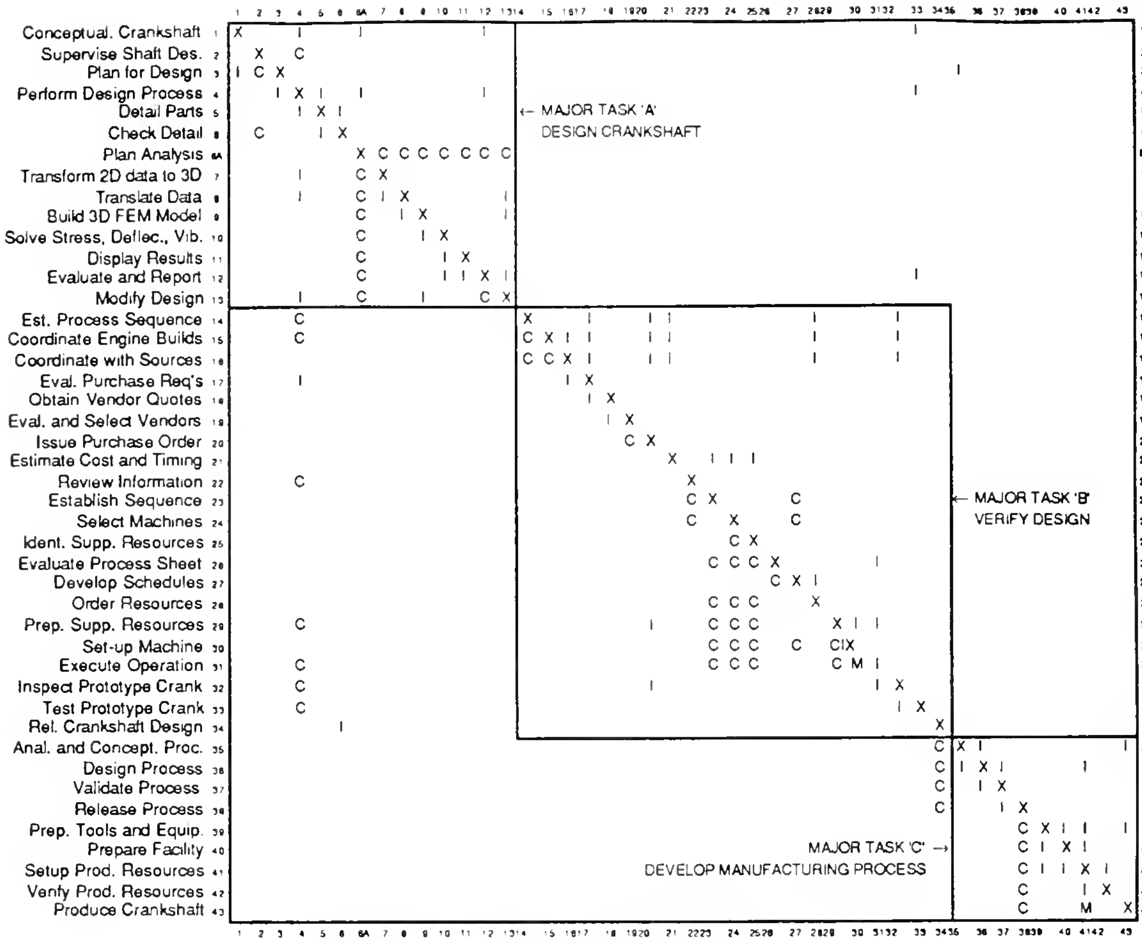

Figure 7. Crankshaft Design Matrix

The matrix allowed us to make some observations about the crankshaft design activity and about its modeling effort. Primarily, the matrix revealed some inconsistencies in the IDEF0 models. In constructing the matrix, certain precedences were inconsistently documented. This confirms the assertion that IDEF0 models are difficult to verify and cross check. Furthermore, application of some simple re-sequencing rules demonstrated that the precedence sets were incomplete. For example, Task 34, Release Crankshaft Design, can be scheduled immediately following Task 6, Check Detail. The precedence shown allows the design to be released before modifications or verifications occur.

The information set contained in the documents was complete enough to allow us to identify the three major activities of the design process and these have been marked as blocks in the matrix. It is of particular interest that the three major tasks of this design activity can proceed almost in parallel with only a few marks linking them at certain stages. These few marks can be considered design drivers which play a critical role in the implementation of improved design procedures such as

design for manufacturing. The implications of this structure and the strategies for managing them are discussed in detail in [5], and are summarized at the end of this section.

Camshaft Design

In a final exercise, complete design and production activities for camshaft development were documented down to similar levels of detail. The previous model for the crankshaft included only the design components of the process, whereas this model expands the steps both before and after design. This camshaft matrix was constructed from the IDEF0 charts exactly as described previously for the crankshaft model. In this more complete model, it was possible to make many more observations and comments regarding the representation and about the process it reflected. The matrix is reproduced in Figure 8A.

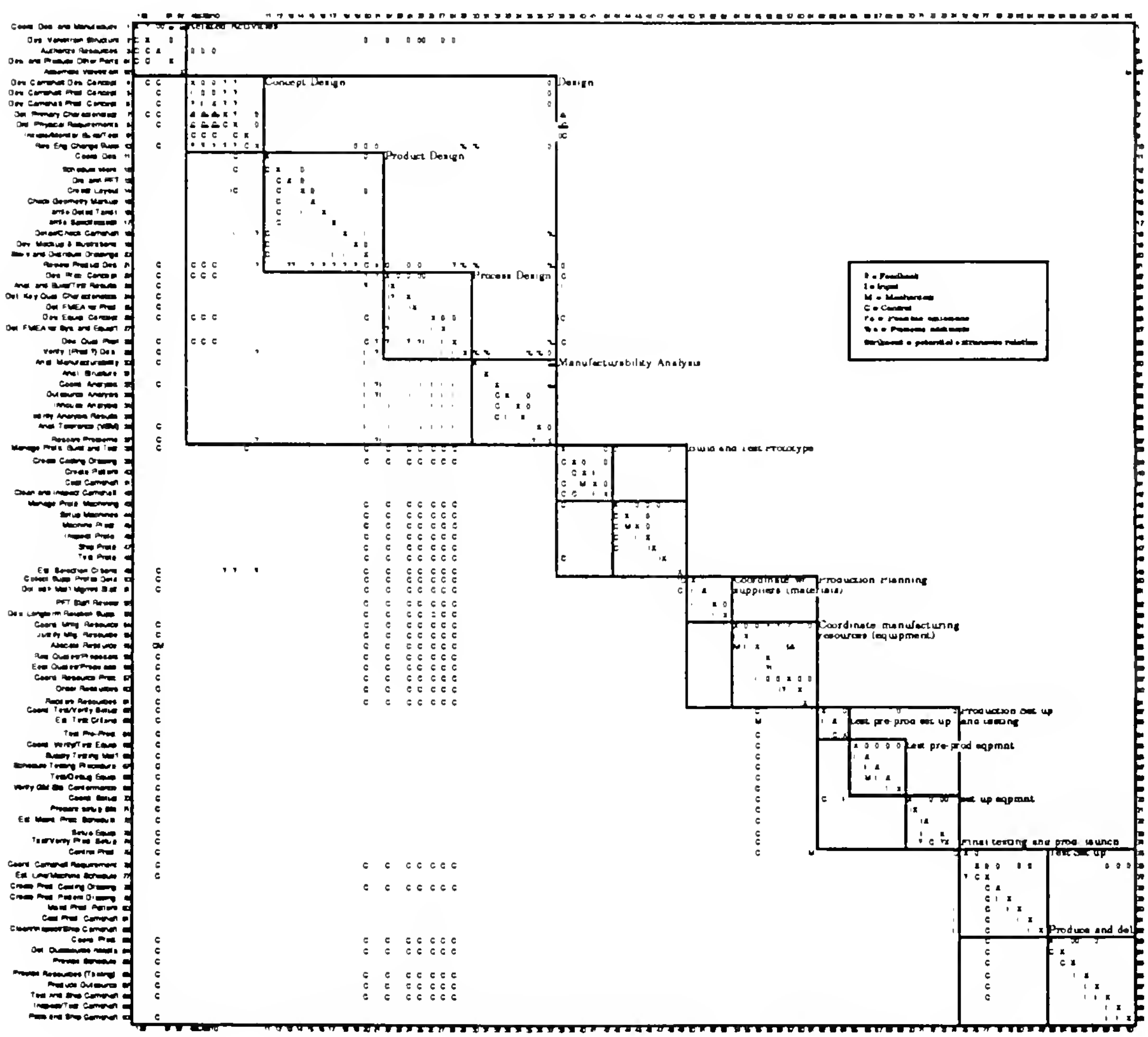

Figure 8A. Camshaft Development Matrix

First, notice that the design activity (Tasks 4-37) is by far the most iterative of the major tasks. Much more communication is required in this stage of the activity than in the production stages which appear to proceed with more sequential tasks (the band of marks just below the diagonal represents a sequential flow). The design activity from this matrix is enlarged in Figure 8B.

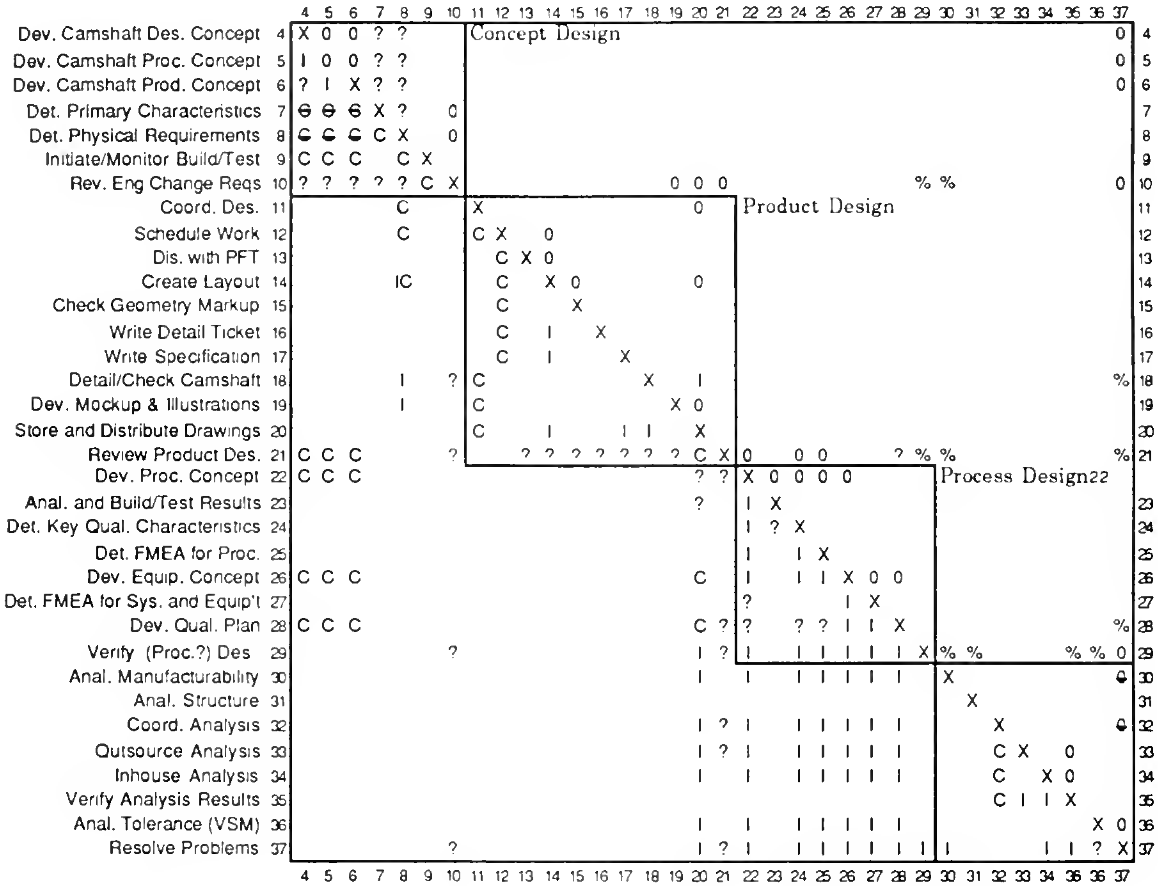

Figure 8B. Design Block of Camshaft Development Matrix

Again, the precedence information was discovered to be incomplete, but in this case, we ventured to classify some of the errors. We identified precedences which should have existed, but were not present in the IDEF0 model. These were conjectured based on the task descriptions and the flow of information expected to be required. For example, task 21 Review Product Design should have input from several Product Design tasks. A number of these possible omissions are denoted in the matrix using the '?' symbol.

We also identified extraneous precedences which seemed superfluous and labeled these as marks with a strikeout. Although these were based on a limited exposure to the details of the process being

modeled, they were based on an engineering interpretation of the task definitions and the expected sources of inputs and destinations of outputs. For example, we question whether determination of primary characteristics and physical requirements of the camshaft (Tasks 7 and 8) require the development of the design and production concepts (Tasks 4, 5, and 6). We suspect that the relation should be modeled in reverse. That is, the determination of the primary characteristics and physical requirements should drive the design of the product and the process.

The remaining marks in the matrix are suggestions for additional feedback. In contrast to the '?' labels which represent a logical relation which may have been overlooked in the modelling, the '%' marks represent opportunities where strategic feedback does not yet exist, but could be implemented. These include suggestions such as including manufacturability analysis (Task 30) in the review of the product design (Task 21).

The most significant shortcoming exposed in constructing these models was the degree of the interdependence between tasks. In many cases, the IDEF0 model denoted an interdependence which seemed more appropriately termed a feedback or verification dependence rather than a tight interdependence which is resolved in an iterative process of negotiation. For example, in the crankshaft model, we expect to conceptualize the manufacturing process, design it, and then validate it (Tasks 35-37). However, their precedence relation indicates that they are executed simultaneously. It makes sense to do it in the present order, but not the other way around. What is needed is an explicit means of differentiating between the various types of interdependent relationships. In this case, we expect that there is some feedback from the design tasks to the conceptualization tasks, but not complete precedence. This type of information is important to capture in the design model and a means for doing so is suggested below.

These matrix models of actual design and manufacturing activities display the critical dependencies explicitly. When information is required before it is available, the precedence appears as an above-diagonal element. The insight provided in these matrix models allow the design manager to choose which dependencies to address based on the potential for overall improvement. It may be possible to eliminate some iteration by re-ordering the activities or by building coordination into the early steps of the process. These and other strategies are discussed in detail in [6], and are summarized below.

Matrix Analysis Tools

The tools available for analyzing design matrices are detailed in [7], but are summarized here. One of the first analysis techniques which the matrix representation facilitates is a re-sequencing of design activities using precedence relations as a guide. This is referred to as partitioning [18]. This procedure attempts to maximize the availability of requisite information and identifies cases in which not all requisite information can be made available. When this occurs, there is a cycle of information flow which appears as a block centered about the diagonal of the matrix. This procedure identifies which tasks are involved in cycles and are performed iteratively. In exposing the structure of dependence, it also makes explicit the relation between the activities. For example, tasks which must be performed in series are differentiated from from tasks which can occur in parallel. Figure 9 depicts a precedence matrix both before and after partitioning.

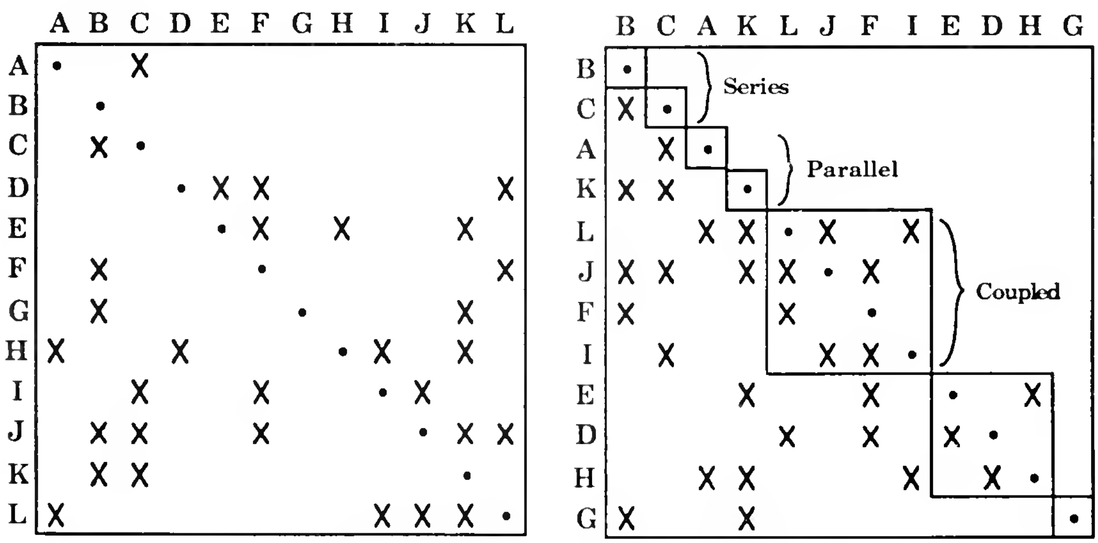

Figure 9. Original and Partitioned Precedence Matrix

Extending the engineering insight can also lead to some design strategies with managerial applications. Task interdependencies within the blocks can be examined to help order activities. Knowing that an activity is in a cycle and will be performed without all of its information, the engineer can evaluate the nature of the interdependence to decide which inputs will best serve to initialize the iterative cycle. This detailed analysis of particular dependencies within iterative cycles is referred to as tearing [10]. The name implies the removal of an arc or dependence to initialize the cycle. For example, in the matrix in Figure 9 above, the engineer might evaluate the relationship between activities E, D, and H to determine which dependence to tear.

The matrix representation easily identifies opportunities for accelerating the design process using a concurrent engineering approach. The precedence information contained in the matrix immediately exposes the earliest start times for each of the activities relative to one another. Certain dependencies may be eliminated by upstream coordination in which the interface between two activities is agreed upon. We call this artificial decoupling [6] which is intended to reduce long iterations found in design activities. For example, if the nature of the dependence between activities E and H above is such that it can be coordinated upstream, the iterative block is reduced to three serial tasks. Strategically decoupling the design into smaller sub-tasks can reduce the size of the working design groups, and can have a dramatic impact on development performance [1, 14].

A related issue involves putting more feedback into the matrix, resulting in more inputs and coupling. An increased coupling strategy is the essential basis of simultaneous engineering and design for manufacture (DFM). In the traditional (sequential) design process, depicted by the matrix in Figure 10a, the product designers would perform their design tasks somewhat independently from the manufacturing engineers. In the modern (concurrent) design process, Figure 10b, the practice of DFM mandates that these two activities be performed simultaneously. This is beneficial because the production expertise is brought into the early design stages (often causing much iteration), resulting in designs which are simpler to manufacture. However, the added coupling in the design process in fact slows product development considerably. Advocates of this philosophy would argue that overall design time can still be reduced because the need for later (more lengthy) iteration is therefore lessened. This is particularly true if the feedback from manufacturing engineering to design was indeed present in the original design procedure. This feedback is shown in Figure 10a by the + marks which depict redesign activity addressing the production problems which inevitably arise.

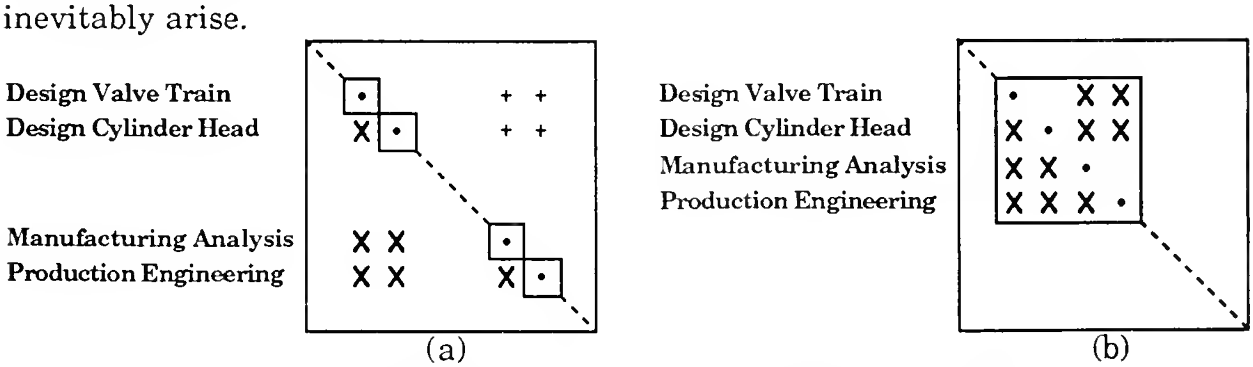

Figure 10. Sequential and Concurrent Design Procedures

Matrix Improvements and Applications

One shortcoming of the representations and techniques presented above is the assumption that all task relations are equal. No attempt is made to quantify amounts of information transferred between tasks in the matrix. It is reasonable to expect that certain dependencies will be stronger than others, or that certain transfers of information will be critical. In the design precedence matrix, task interactions are not differentiated from one another.

We aim to embody some information about the tasks' interdependencies into the matrix itself and to incorporate this dimension into the analysis techniques applied to the design process. The binary design structure matrix can be vastly improved by having the matrix elements reflect any of a number of different measures about the relation between tasks or the task's relation to the entire design process. These numerical measures can reflect quantity of information transferred between tasks, sensitivity of tasks to incomplete data, or other quantities.

One such numerical scoring is based on an estimate of the quantity of information transferred between tasks. This would allow us to differentiate between different degrees of interdependence and would significantly clarify the models. Simple verifications and feedbacks can be differentiated from strict precedence which requires iteration and negotiation. By attaching numerical data to the precedence data, the matrix can represent more information than it presently does. The additional advantage to this quantitative model is the number of analysis tools which can be applied to it. With the enhanced representation comes additional capability to suggest improvements.

For example, a sequence can be scored based on the quantity of information that must be approximated because it is not available when it is required. If a particular ordering requires too much information to be assumed, other sequences can be generated and scored. These scores allow comparisons of different design structures using a common metric. A numerical algorithm can be aimed at minimizing a certain score. The score can be tailored to use a particular metric most appropriately. For example, if one were to measure the certainty with which a particular piece of information could be approximated (perhaps based on previous experience), it would be best to maximize the minimum above diagonal element. These heuristics can easily be changed to accommodate various measures of task interdependence.

An example of a numerical design structure matrix is presented in Figure 11. In this example, a matrix has been constructed to demonstrate the power of numerical analysis techniques. A twelve by twelve precedence

matrix has been converted from binary dependencies to numerical metrics of interdependence. Analysis was performed on the matrix using two methods: (1) standard algorithms which consider all dependencies equal and (2) numerical algorithms which rely upon an enhanced information content.

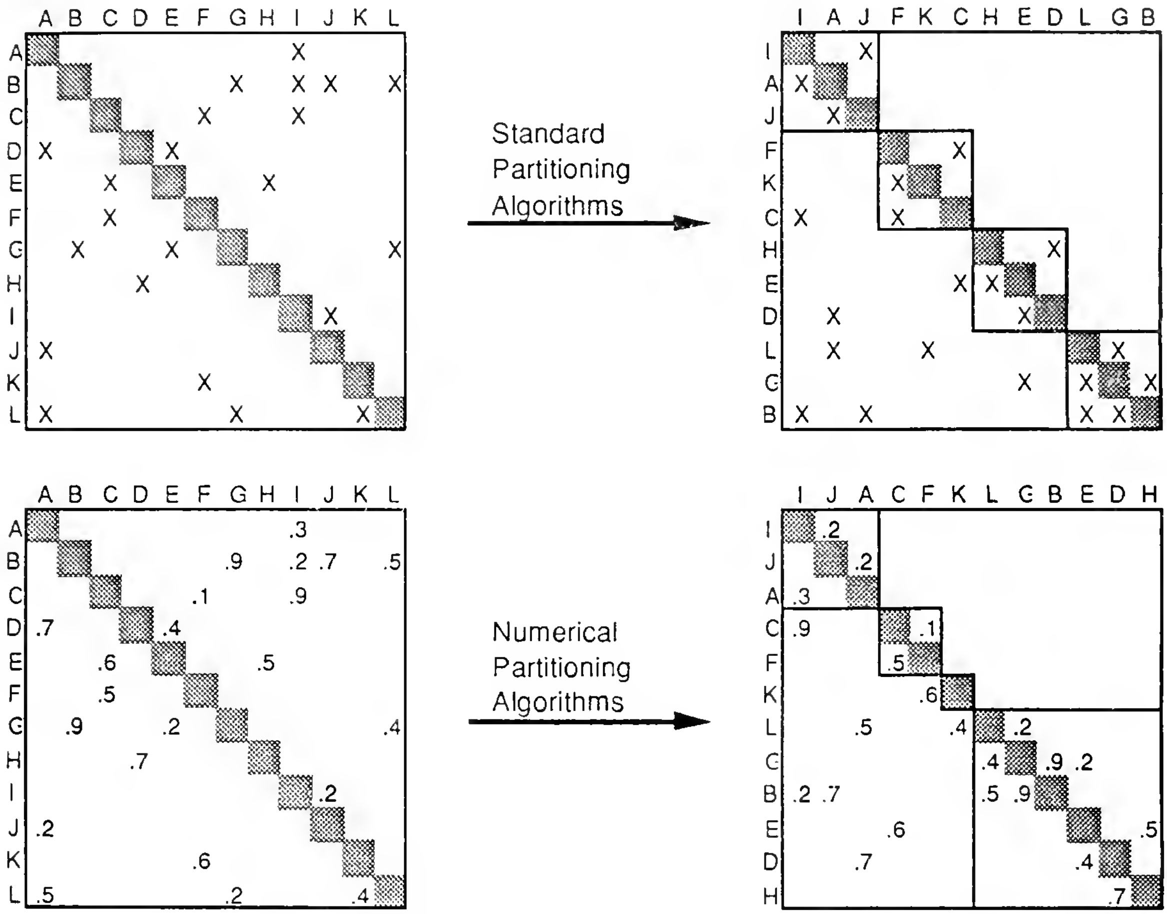

Figure 11. Numerical Design Structure Matrix

In this example, the numerical goal was to minimize the sum of the above diagonal elements weighted by the distance from the diagonal. Compare the results from standard partitioning to the output from the numerical algorithms. The results are significantly different and lead to different conclusions.

The different results suggest different design management strategies. In the binary matrix, Task G's dependence on Task E would not be considered a strategic one which could impact the execution of the design activity. However, the numerical matrix provides additional insight which could lead to a design organization pictured in Figure 12. Recognizing this tear in the binary matrix would have required much

insight or experimentation, while in the numerical matrix, the leverage of this tear is more apparent.

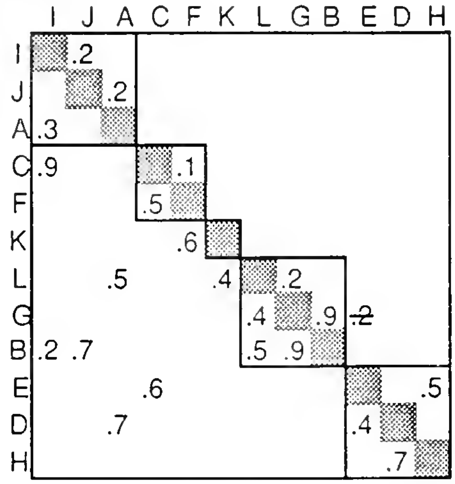

Figure 12. Tear Suggested by Numerical Matrix

The enhanced matrix representation will capture significantly more information in an easily usable form. The enhanced representation capable of handling measures of task interdependence provides enough flexibility to be used in many scenarios and at different levels. A parameter level matrix might measure sensitivity of a parameter to its input. These numerical matrix representations can be analyzed using any number of numerically based tools which utilize the information appropriately to identify both engineering and management improvements.

Conclusion

The work reported here is a comparison of existing design representation techniques. The need for good models has been prompted by an increasing interest in executing and managing the design activity in the most effective manner. As different representations of design have been utilized, limitations with existing techniques have been exposed. Using a precedence matrix to model design activities at General Motors has aided our understanding of the complex interactions which occur during the design process. As the study of the design has become more sophisticated, we have found that enhancements to this representation are necessary. The proposed representation based on a numerical precedence matrix incorporates the qualities of a good representation outlined above. The flexibility and clarity of the numerical matrix format provide a platform from which to address questions which are relevant to improving the design process.

Acknowledgements

This research is funded jointly by several sources including the National Science Foundation, the General Motors Corporation, and the MIT Leaders for Manufacturing Program, a partnership involving eleven major US manufacturing firms and MIT's engineering and management schools.

References

[1] D. G. Ancona and D. E. Caldwell. "Demography and Design: Predictors of New Product Team Performance", *Working Paper, MIT Sloan School of Management.* no. 3078-89, September 1989.

[2] T. A. Black. *A Systems Design Methodology Applied to Automotive Brake Design*, S.M. Thesis, MIT, 1990.

[3] T. A. Black, C. H. Fine and E. M. Sachs. "A Method for Systems Design Using Precedence Relationships: An Application to Automotive Brake Systems", *Working Paper, MIT Sloan School of Management.* no. 3208-90, October 1990.

[4] J. Corbett. "Design for Economic Manufacture", *Annals of C.I.R.P.* vol. 35, no. 1, pp. 93, 1986.

[5] S. D. Eppinger. *Model-Based Approaches to Managing Concurrent Engineering.* International Conference on Engineering Design, Zurich, 1991.

[6] S. D. Eppinger, D. E. Whitney, R. P. Smith and D. A. Gebala. *Organizing the Tasks in Complex Design Projects.* ASME Design Theory and Methodology Conference, Chicago, IL, 1990.

[7] D. A. Gebala and S. D. Eppinger. "Methods for Analyzing Design Procedures", *Working Paper, MIT Sloan School of Management.* no. 3280-91, April 1991.

[8] F. Harary. *Graph Theory.* Addison-Wesley, Reading, Mass., 1969.

[9] D. M. Himmelblau. "Decomposition of Large Scale Systems, Part 1: Systems Composed of Lumped Parameter Elements", *Chemical Engineering Science.* vol. 21, pp. 425-438, 1966.

[10] G. Kron. *'Diakoptics', Piecewise Solution of Large Scale Systems of Equations*, Ph.D. Thesis, University of Texas, Austin, 1963.

[11] W. P. Ledet and D. M. Himmelblau. "Decomposition Procedures for the Solving of Large Scale Systems", *Advances in Chemical Engineering.* vol. 8, pp. 185-254, 1970.

[12] F. K. Levy, G. L. Thompson and J. D. Wiest. "The ABC's of the Critical Path Method", *Harvard Business Review.* vol. 41, 5, pp. 98 - 108, 1963.

[13] D. A. Marca and C. L. McGowan. *SADT: Structured Analysis and Design Technique.* McGraw Hill, New York, 1988.

[14] J. B. Quinn. "Managing Innovation: Controlled Chaos", *Harvard Business Review*. pp. 73-84, May-June 1985.

[15] J. L. Rogers. *DeMAID: A Design Manager's Aide for Intelligent Decomposition User's Guide*. NASA Technical Memorandum 101575, March, 1989.

[16] D. T. Ross. "Structured Analysis (SA): A Language for Communicating Ideas", *IEEE Transactions on Software Engineering*. vol. SE-3, no. 1, January, pp. 16-34, 1977.

[17] D. T. Ross. "Douglass Ross Talks about Structured Analysis", *IEEE Computer Magazine*. 1985.

[18] R. W. H. Sargent and A. W. Westerberg. "'Speed-Up' in Chemical Engineering Design", *Trans. Instn. Chem. Engrs*. vol. 42, T190-T197, 1964.

[19] M. Sequeira. *Use of the Design Structure Matrix in the Improvement of a Vehicle Development Process*, Master's Thesis, MIT, 1991.

[20] D. V. Steward. *Systems Analysis and Management: Structure, Strategy, and Design*. Petrocelli Books, New York, 1981.

[21] D. V. Steward. *Using Information Flow to Manage the Design of Systems*. Portland International Conference on Management of Engineering and Technology, Portland, Oregon, October, 1991.

[22] J. N. Warfield. "Binary Matrices in System Modeling", *IEEE Transactions on Systems, Man, and Cybernetics*. vol. SMC-3, no. 5, September, pp. 441-449, 1973.

[23] J. D. Wiest and F. K. Levy. *A Management Guide to PERT/CPM*. Prentice-Hall, Englewood Cliffs, New Jersey, 2nd Edition, 1977.

[24] P. H. Winston. *Artificial Intelligence*. Addison-Wesley, Reading, Mass., 1984.

6809 077